DOCUMENT RESUME

ED 445 893 SE 063 722

AUTHOR	Stratford, Steven J.
TITLE	A Review of Computer-Based Model Research in Precollege Science Classrooms.
SPONS AGENCY	National Science Foundation, Arlington, VA.
PUB DATE	1997-00-00
NOTE	13p.
CONTRACT	RED9353481
AVAILABLE FROM	For full text: http://hi-ce.org/hiceoffice/papers/papers/review-of-computer based-model/index.html.
PUB TYPE	Information Analyses (070)
EDRS PRICE	MF01/PC01 Plus Postage.
DESCRIPTORS	*Computer Simulation; *Computer Software Development; *Computer Uses in Education; Educational Research; Educational Technology; High Schools; Instructional Materials; Learning Processes; *Models; Science Education; *Scientific Concepts; Scientific Methodology

ABSTRACT

Computers, because of their computational speed and multi-representational capabilities, are particularly well-suited to the implementation of dynamic models. This paper reviews research conducted on the use of computer-based models in precollege science classrooms. Topics addressed include Investigations of Running Simulations, Investigations of Modeling, Investigations of Writing Simulation Programs, and Future Directions. (Contains 41 references.) (WRM)

Reproductions supplied by EDRS are the best that can be made
from the original document.

A Review of Computer-Based Model Research in Precollege Science Classrooms

Steven J. Stratford

School of Education, Department of Educational Studies

University of Michigan, Ann Arbor, MI 48109 USA

S. Stratford

U.S. DEPARTMENT OF EDUCATION
Office of Educational Research and Improvement
EDUCATIONAL RESOURCES INFORMATION CENTER (ERIC)
This document has been reproduced as received from the person or organization originating it.
Minor changes have been made to improve reproduction quality.
Points of view or opinions stated in this document do not necessarily represent official OERI position or policy.

1

Introduction

ED 445 893

As microcomputers attain a ubiquitous presence in precollege classrooms, science educators turn to software based on scientific models to help their students construct rich understandings of scientific concepts. In science, models are designed to imitate natural systems, capturing the essence of major components and mimicking their interactions (Kornblugh and Little, 1976). The scientific endeavor itself has been described as a process of constructing models for their conceptual and predictive value (e.g., Stewart & Hafner, 1991 or Gilbert, 1991). Black argued long ago (1962) that understanding models should be integral to scientific practice; of particular value in promoting learning are analog models that abstractly represent key structures or patterns of relationships within the modeled object (for example, ball-and-stick formations representing the crystalline structure of chemicals). Forrester (1968), who could rightly be called the father of dynamic (time-based) system modeling, claimed that creating and running dynamic models should help clarify one's own mental models and foster deeper understanding of complex systems. Gee (1978) noted the parallel between model-creation as a heuristic for scientific theory development and the model itself as a pedagogical tool for the intellectual growth of the learner. Recent science education reform efforts such as the American Association for the Advancement of Science's Project 2061 (1993) suggest that in pre-college classrooms, "Mathematical models and computer simulations are used in studying evidence from many sources in order to form a scientific account of the universe." Computers, because of their computational speed and multi-representational capabilities, are particularly well suited to the implementation of dynamic models. Hence, the topic of this research review column is computer-based models in precollege classrooms.

Table 1 lists terms and definitions associated with the rather broad phrase `computer-based models,' as used in this column. `Running a simulation' refers to operating or executing a computer program that is based upon a scientific model. `Modeling' refers to the act of creating or revising a model; in other words, modeling is using a `software modeling environment' to build models without having to learn to `write a simulation program.'

Term	Meaning
model	a scientific construct designed to imitate a real-world phenomenon
simulation	a computer program based on a model
writing a simulation program	coding, testing, and debugging a simulation using a general purpose programming language
running a simulation	executing a simulation in order to observe its behavior and draw inferences about the phenomenon it models
modeling	the act of creating or revising a model
modeling environments	computer programs whose purpose is to allow the user to create computer-based models without actually writing a computer program

Table 1. Terms associated with the phrase `computer-based model'.

Many simulations have been written specifically for precollege classroom use. In most of these simulations, students are restricted to varying parameters and observing effects--the underlying model is usually inaccessible. Allowing students to construct or revise models has been impracticable because few age- or ability-appropriate

SE 063 727

BEST COPY AVAILABLE

modeling tools existed for precollege students. Today, however, newer and more learner-friendly software modeling environments enable students not only to examine how models work, but to construct models of their own.

While both building models and running simulations are learning activities compatible with the constructivist perspective, each provides somewhat different learning opportunities. The skills necessary to create one's own model are quite different from those required to run a simulation based on someone else's model, as are the potential learning outcomes. Each of the three activities under consideration in this column (running simulations, creating simulation programs, and creating models) tends to support different research questions, because they vary according to cognitive demand (in the form of prior knowledge and skills required for the activity) and according to the kinds of operations that must be mastered in order to succeed. Table 2 is a simple attempt to elaborate upon the ways in which they differ.

Type of Learning Activity	Cognitive Demand	Operations Students Must Master
Running a simulation	Scientific domain	Domain-specific
Creating a model using a modeling environment	Scientific and modeling domains	Domain plus generic modeling
Creating simulation programs with general-purpose programming languages	Scientific, modeling, and programming domains	Domain plus modeling plus low-level programming

Table 2. Levels of cognitive demand and abstraction of model-based learning activities.

A simulation focuses attention on the reality it purports to model. Some researchers have observed student interactions with simulations to discover the extent to which the simulation's real-world-imitating behavior confronts students with their (mis)conceptions of reality. Other researchers have investigated students' problem-solving strategies as they solve problems using realistic simulation-generated data. Modeling, on the other hand, requires students to master modeling concepts, and may require substantial prior domain knowledge in order to build meaningful models. Researchers using modeling software have focused on issues such as the feasibility of having students construct models in classrooms, the understandings they construct about models and modeling, and the cognitive benefits of thinking about dynamic systems. Finally, in order to write a simulation program, students must be engaged at a high level of cognitive load since they must have some understanding not only of the domain and of modeling, but also of a programming language as well. Investigations of students writing simulation programs have been concerned with student learning and motivation in general.

Few reviews of the literature on simulation and modeling research have been published, perhaps owing to the scarcity of studies, the difficulty of research, and the infancy of the field. In this column, I review recent studies (within the past 7 years) in terms of their contributions to the literature on computer-based models in precollege classrooms. Table 3 presents a summary of the work reviewed, in the order they appear in this column, organized into three categories: running simulations, modeling, and writing simulation programs. The table contains, for each study, the name of the software, the main purpose of the study, and its contribution to the literature.

Software:Reference	Study Purpose/Theory	Study Contribution
Running Simulations		
Brna (1987, 1991): DYNLAB	Software presented a variety of situations which were expected to cause a student to confront discrepancies between their beliefs about motion and the behavior of their model	The simulation provided opportunities for students to change their conceptual understanding and to articulate their new beliefs.

Gorsky & Finegold (1992): 5 force simulations	Investigated the use of computer simulations to help restructure students' conceptions of force	Simulations led to varying degrees of cognitive dissonance and were effective in eliciting students' beliefs about forces acting on objects at rest and in motion. Students who directly experienced the outcomes of their own misconceptions apparently rejected their incorrect views and accepted the scientific ones, at least in the context of the simulation.
White (1993): ThinkerTools	Described a physics motion simulation used in an inquiry curriculum, designed to help students develop conceptual knowledge	Sixth graders performed much better on classic force & motion problems than high school students in traditional physics classes.
Slack & Stewart (1990): GCK	Explored individual students' problem-solving strategies and developed a model of student performance; GCK genetics simulation was used to present problems and data to students	Students followed these strategies: unplanned approach (lack of hypotheses & testing strategy); working backward (explaining rather than predicting); and emphasizing quantitative counting and ratios. Students lacked problem-solving abilities and skills such as genotypic thinking and generational thinking.
Hafner (1991); Stewart, Hafner, Johnson, & Finkel (1992): GCK	Used GCK to investigate individual students' model-revising processes, general and domain-specific heuristics, and criteria for model acceptance	Students engaged in model-revising problem-solving successfully, and were able to produce revisions which were generally compatible with accepted scientific theory. The simulation allowed students to engage in knowledge production, and significantly increased the amount of "research" they could do.
Finkel (1993, 1994); Finkel & Stewart (1994): GCK	Studied how model-revision strategies and knowledge were used as students worked in groups to solve genetics problems using the GCK simulation	Students' strategies for model revision included a variety of actions such as recognizing anomalous aspects of the data, making crosses, and developing, assessing and accepting multiple alternative models. Students used their understanding of genetics, of the process of model revision, and of their own problem-solving strategies during model revision.
Simmons & Lunetta (1993): CATLAB	Explored general patterns of problem-solving behaviors and genetics conceptual organizers in experts and novices interacting with a genetics simulation	Successful expert and novice problem solvers employed the most complex patterns of problem-solving behaviors, mainly using description problem-solving sequences; least successful employed random patterns of behaviors.

Ronen, Langley, and Ganiel (1992): STEP simulations	Reported and analyzed a large scale integration of computerized simulations into the present structure of Israeli high schools	Authors speculated that problems encountered were symptoms of problems which occur in systems in transition. They also suggested that real change can only occur after teachers experience the advantages offered by computer simulations.
Kruper & Nelson (1991): Biota	The software allowed students to construct meaning by providing opportunities to define problems, construct and test alternative hypotheses, and communicate subsequent evaluation of these hypotheses to peers	They reported no significant differences on pre- and post-test between treatment and control groups on tests of science reasoning skills, however, there were differences between learning processes. They concluded that strategic simulations can offer students valuable experiences which help develop deeper content understanding.
Feurzeig (1992): Cardio	Provided an interactive visual environment for investigating the physiological behavior of the heart. The simulation incorporates process visualization aids in the introduction of model-based inquiry skills, and supports advanced work in science research	Students were able to explain nonlinear dynamical behavior and to solve heart repair problems.
Richards, Barowy & Levin (1992): Explorer Science	Provided students with a coherent set of experiences that challenge the way they think about the world; provided opportunities to construct and test explanations for phenomena	Students developed a sense of how scientists use models. Authors reported that student interaction with simulated models facilitated analysis and conceptual understanding of physical phenomena.
Creating Models		
Jackson, Stratford, Krajcik, and Soloway (1995): Model-It	Described intentional scaffolding strategies designed to make system dynamics modeling accessible to pre-college students	Students built reasonable models; software strategies made modeling accessible; building models allowed students to refine and articulate their understanding of complex systems.
Mandinach (1988): STELLA	Investigated the effectiveness of using STELLA in the systems-thinking curricula	Students tested well on their knowledge of STELLA, but were less able to translate knowledge and skills to more general problems.
Mandinach (1989): STELLA	Tested the potentials and effects of using STELLA to teach content-specific knowledge as well as general problem solving skills	Students acquired knowledge of systems concepts and applied them to scientific problems at varying levels of complexity and sophistication.
Mandinach & Cline (1992): STELLA	Examined the impact of learning from a systems thinking approach to instruction and from using simulation-modeling software	The authors concluded from their experiences and observations that gaining a working knowledge of system dynamics, STELLA software, and the Macintosh is substantially different from acquiring information within a content area of expertise.

BEST COPY AVAILABLE

Miller, Ogborn, Briggs, Brough, Bliss, Boohan, Brosnan, Mellar, and Sakonidis (1993): IQON	Authors described the design of and rationale for modeling tools that are claimed to be simple enough for young teenaged students (grade 8) to learn	Pupils built meaningful models of considerable complexity and contributed ideas about the relation of IQON models to reality. Authors observed that students began to understand complex models as interconnected systems. Pupils constructing models saw their models as fallible, tended to consider revisions, and made more interesting modifications than those who were simply exploring pre-defined models.
Schecker (1993): STELLA	Research focused on having students develop and test models with STELLA; author suggested that modelling can help to accentuate the conceptual structure of a physical domain and help clarify the qualitative meaning of physical concepts	It took about 2 instructional units for students to become familiar with software to make models on their own, after which they were able to work out model structures themselves in classroom discussions or work groups.
Creating Simulation Programs		
diSessa (1991): Boxer	Investigated ways in which sixth grade students invented ways of working on difficult problems	Students engaged in student-initiated learning (they learned to "cheat" at the simulation in order to solve difficult problems).
Fuertzeig (1992): Function Machines	Investigated the use and benefits of visualization in model-based inquiry activities	The author suggested that appropriate computer modeling activities can make the experience of doing science concrete and highly motivating for high school students.
Guzdial (1995): EMILE	Created a scaffolded environment in which helped students to create physics simulations in Hypercard without learning to program first	Students learned about programming, and learned physics concepts (velocity, acceleration, projectile motion) through creating simulations in Hypercard.

Table 3. Studies of computer-based modeling within the past 7 years.

Review of Research

Investigations of Running Simulations

The computer's ability to model Newtonian motion has prompted several researchers to study how student interactions with simulations might lead to conceptual change. Brna (1987, 1991), Gorsky and Finegold (1992) and White (1993) all suggested, using varying terminology, that computer simulations can help students identify and revise scientifically inconsistent conceptual understandings.

• Brna (1987, 1991) reported on research using DYNLAB dynamics and kinematics simulation software. In these studies, the researcher designed a set of physical situations which would create conceptual difficulties for the students, on the assumption that learning is promoted through confronting students with inconsistencies in their beliefs. In DYNLAB, students guide an object through a pre-determined path by giving the object a series of instructions. In the 1987 study, students took a pre-test on kinematics, received instruction on the operation of the software, and then worked for up to 2 hours to solve one or more of the pre-test problems with DYNLAB, whereas the 1991 study presented a single case study of student problem solving. Brna reported, using anecdotal data, that in the course of using the software, students were confronted with their misconceptions, and some of these confrontations were resolved. He also indicated that those who took advantage of the confrontations were often the

ones who were eventually able to articulate their own beliefs. The author conjectured that one benefit of DYNLAB was that it could help teachers distinguish between students with Newtonian beliefs and those with non-Newtonian beliefs.

• Gorsky and Finegold (1992) used a series of simulations involving forces on objects (e.g. book at rest on table, pendulum bob at rest, book sliding on frictionless table) in which students could indicate directions of forces on the object, and then watch the resulting action. The behavior of the object might be incompatible with real world behavior, and these "discrepant events" generate cognitive dissonance. The researchers found that simulations led to varying degrees of cognitive dissonance and were effective in eliciting students' beliefs about forces acting on objects at rest and in motion. They also indicated that students who directly experienced the outcomes of their own misconceptions apparently rejected their incorrect views and accepted the scientific ones, at least in the context of the simulation. They hypothesized that using such simulations prior to classroom instruction may help teachers deal with students and their beliefs on an individual basis.

• White's (1993) ThinkerTools curriculum features a series of force-and-motion computer microworlds embedded in a sixth-grade curriculum of science inquiry. (I discuss the inquiry curriculum below.) Each microworld has a goal; for example, one goal might be to get a dot to hit a target with a given velocity. She hypothesized that in order for students to gain scientifically sound conceptual models of force and motion, students' knowledge must not only undergo a change in form but also in content, a process of conceptual change facilitated by the software. The curriculum and software was used with two sixth grade science classes for two months. On a final test of conceptual physics knowledge, the sixth graders in the treatment group performed better on classic force and motion problems than other sixth graders who did not receive the curriculum treatment and better than high school students in traditional physics classrooms.

Simulations based on genetics models can generate realistic problems for students to solve and realistic data for them to analyze. A number of papers (Slack & Stewart, 1990; Stewart & Hafner, 1991; Hafner, 1991; Stewart, Hafner, Johnson, and Finkel, 1992; Finkel, 1993 & 1994; Finkel and Stewart, 1994; and Simmons and Lunetta, 1993) described systematic investigations into identifying students' problem-solving behaviors while using genetics simulations.

• Slack and Stewart (1990) used the Genetics Construction Kit (Jungck & Calley, 1993) to add to understanding of individual students' problem-solving strategies and to develop a model of student performance. The Genetics Construction Kit (GCK) simulates the Mendelian genetics model; however, it should more accurately be called a strategic simulation (Jungck & Calley, 1985) because decisions as to which organisms to cross are fully under the control of the user, not time-based and computer-controlled. In Slack and Stewart's study, thirty-five students solved genetics problems which required them to reason from effects (phenotype data) to causes (genotype data); data consisted of think-aloud protocols and computer-generated information on the problems presented and crosses performed. The authors reported that students followed these problem-solving strategies: an unplanned approach (lack of hypotheses and testing strategy), a working backward approach (explaining rather than predicting), and an approach emphasizing quantitative counting and ratios. Students lacked problem-solving abilities and skills such as genotypic thinking and generational thinking, focusing instead on phenotypical interpretations of data. The authors concluded that computer simulations which provide a realistic problem-solving environment are still not sufficient to elicit good problem-solving skills because the simulation does not help students develop connections between conceptual knowledge and problem-solving strategies. They provided a number of recommendations for genetics instructional design, such as explicitly teaching hypothesis generation and testing strategies and presenting genetics concepts and principles so that relationships between concepts are obvious.

• In Hafner (1991) and Stewart, Hafner, Johnson, and Finkel (1992) the authors described a high school course in which students solved genetics problems over an extended time period using GCK. They set out to describe individual students' model-revising processes, the general and domain-specific heuristics they used, and the criteria they used to decide if a model is acceptable. Six students used GCK to solve up to seven different kinds of genetics problems; students were asked to think aloud as they worked. The researchers found that students were able to engage in model-revising problem solving successfully and were able, starting with simple models, to produce revisions of increasing complexity that were generally compatible with accepted scientific theory. The authors claimed that one advantage of simulations such as GCK is that it allows students to engage in knowledge production in the classroom, significantly increasing the amount of "research" they can do, as compared to actually crossing fruit flies in the laboratory and observing their offspring.

• Finkel (1993, 1994) and Finkel and Stewart (1994) observed groups of students, also solving problems with GCK. The genetics problems posed by the computer required effect-to-cause reasoning; i.e. the outcome of a cross was presented to the students, and they worked to revise the Mendelian model to explain anomalous data. Of

interest were students' model-revising problem-solving strategies. Finkel found that students' model-revision processes included actions such as recognizing anomalous aspects of the data, making crosses, and developing, assessing and accepting multiple alternative models. In addition, she reported that students used their knowledge of genetics, knowledge of model-revising process, and knowledge of their own problem-solving strategies during model-revising problem-solving.

• Simmons and Lunetta (1993) explored the general patterns of problem-solving behaviors and genetics conceptual organizers in experts and novices interacting with a computer simulation called CATLAB. CATLAB is a genetics simulation that requires students to generate questions and hypotheses, choose control variables, gather and interpret generated data, and make inferences and draw conclusions. The researchers asked 3 experts and 10 high school biology students to solve genetics problems with the software. They found that successful problem solvers employed the most complex patterns of problem-solving behaviors, principally using description problem-solving sequences; the least successful problem solvers employed random patterns of behaviors.

Two studies discussed curricular concerns related to using computer-based models in the classroom. Both White (1993) and Ronen, Langley, and Ganiel (1992) emphasized the important role of the teacher in their classroom curriculum innovations.

• White (1993) integrated the ThinkerTools software into an instructional cycle which consisted of a four phases: motivation, model evolution, formalization, and transfer. Each phase involved collaboration and discussion, sometimes in small groups at the computer, sometimes as a class with the teacher leading. In the motivation phase, students predicted outcomes of simple real-world force and motion situations. Then, in the model evolution phase, they performed experiments in a computer microworld designed to mimic the real-world situation. Students next constructed laws to describe the behavior of the microworld during the formalization phase. Finally, in the transfer phase, students applied their formalized law to see how well it predicted the behavior of the initial real-world situation. They also conducted experiments to verify their predictions (and also to compare its behavior to the microworld's behavior). The author suggested that in order for this inquiry-based curriculum to be successful, teachers must understand the inquiry process and the purpose of each activity in order to scaffold students' discussion and collaboration activities.

• Ronen, Langley, and Ganiel (1992) reported on a curriculum project in Israel to integrate the STEP simulations into 18 schools (18 teachers, 50 physics classrooms, 300 lessons). They evaluated the success of their efforts using teacher-generated reports on the results of each lesson, teacher and student final assessments, and reports on class observations and personal interviews. Teachers felt the simulations contributed favorably to the subject matter, and improved student interest and involvement. However, there was no agreement as to the best use of simulations in the classroom: teachers suggested that simulations could be used for a wide variety of purposes: "initial subject presentation," "exploration during teaching," "drill and practice," "assisting laboratory work," or "summary and review." Most students felt the simulation contributed toward a better understanding of the subject. However, the researchers reported that the implementation was hindered by logistic difficulties (e.g. many computers were located in computer labs not science classrooms), by time and program constraints (e.g. teachers were reluctant to take time away from students' preparation for final examinations), and psychological/didactical issues (e.g. teachers used the simulation for canned demonstrations instead of student inquiry and exploration). The authors suggested that many of these problems were only superficial excuses, symptoms of deeper problems related to an educational system in undergoing change. They suggested that in order for change to occur, teachers must realize the computer's potential and limitations; the key to helping teachers realize this is to provide opportunities for them to personally experience the advantages of these new tools over existing methods.

Finally, three studies reported results on student understanding of scientific concepts. Two of these studies investigated the pairing of real-world demonstrations with corresponding simulations. Kruper and Nelson (1991) reported that they found very little difference in science reasoning skills between those using simulations and those not, while Richards, Barowy and Levin (1992), on the other hand, reported that tightly coupling a real-world demonstration with a simulation helped students learn about modeling and about science content.

• Kruper and Nelson (1991) reported on a biology laboratory study using Biota, another BioQUEST simulation (Jungck & Calley, 1993). Biota is a strategic simulation of processes influencing sizes of plant and animal populations. They compared students in the traditional "wet lab" environment with the simulation environment, and focused on the development of science reasoning skills and on differences in the learning environments. They found that students in the Biota lab were able to collect, analyze and interpret data several times in the same time it took the wet lab group to perform one experiment. They reported no significant differences on pre- and post-test between treatment and control groups on tests of science reasoning skills, and suggest that simulations such as Biota are most effective when they are used as part of a progressively deeper domain study rather than in an

isolated 2-hour laboratory session.

• Feurzeig (1992) described a simulation called Cardio, which provides an interactive visual environment for investigating the physiological behavior of the heart, while enabling students to gain insight into the dynamics of oscillatory processes. Students explored the simulation, then engaged in several problem-solving activities. Feurzeig reported that students were able to explain nonlinear dynamical behavior of the heart without being explicitly instructed on the phenomena and were also able to solve heart repair problems.

• Richards, Barowy, and Levin (1992) reported a formative evaluation on a simulation called Explorer, an interactive environment which uses animated computer models and incorporates data analysis tools (the environment includes graphs, spreadsheets, scripting, and interactive tools). In classroom user testing, students compared the behavior of simulated bouncing balls with the behavior of actual physical balls. Using anecdotal data, they reported that by tightly coupling real-world observations with simulations, students were able to investigate a phenomenon in ways which would otherwise be beyond the scope and abilities of high school students.

Overall, the literature reviewed here indicates that simulations are useful in confronting students with their misconceptions in order to promote conceptual change. Investigations into the extent to which simulations can affect students' acquisition of conceptual knowledge appear to be promising. Simulations also appear to be useful tools in investigations of students' problem-solving strategies and behaviors. Finally, the literature indicates that simulations can be effectively integrated into appropriate theory-building and inquiry activities, although large-scale integration into school systems has proven to be difficult.

Investigations of Modeling

A number of studies have emphasized that software can be used to make modeling "accessible" to students in precollege classrooms. Miller, Ogborn, Briggs, Brough, Bliss, Boohan, Brosnan, Mellar, and Sakonidis (IQON; 1993), Schecker (STELLA; 1993), and Jackson, Stratford, Krajcik, and Soloway (Model-It; 1995) all claim that modeling environments make model creation more accessible to students, especially in contrast to previously available software that is designed for experts, hard to use, or lacking a user-friendly interface.

• Jackson et al. (1995a, 1995b) reported on software called Model-It, a modeling environment based upon ecological modeling techniques (Silvert, 1993). The environment provides software-implemented scaffolding intended to ground model building tasks in students' prior knowledge and to bridge their understanding of models toward more abstract and formalized representations. Also, its model-testing capabilities facilitate a close coupling between students' mental models of a phenomenon and how the model's behavior is displayed. Twenty-two students in a project-based science classroom received four days of training and then constructed models for one day. The authors presented qualitative data to show that (1) software-realized scaffolding provided by the software supported model construction, and (2) students created meaningful ecological models of reasonable sophistication and complexity for their grade level.

Several studies of student modeling reported on the effects of modeling environments on student learning. Mandinach and Cline (1988, 1989, 1992, 1994) and Schecker (1993) investigated the impact of curricular innovation involving systems thinking and the STELLA modeling software on student learning and transfer. Miller et al. (1993) described several different modeling tools and presented results on students' reasoning skills.

• In one of the longer-running research efforts involving student modeling, Mandinach and Cline (1994) have engaged in an investigation of the dynamics of implementing a technology-based learning environment centered around systems thinking using STELLA (e.g. Roberts et al., 1983 or High Performance Systems, 1992). STELLA is a Macintosh-based tool for constructing system dynamics models (Forrester, 1968; High Performance Systems, 1992). Early research goals focused on students' mastery of systems thinking, on the potential impact of systems thinking upon learning outcomes and transfer, and on the effectiveness of STELLA as a learning tool. They concluded (Mandinach, 1988) that systems thinking and STELLA affected learning and teaching activities, but in different ways for different teachers and disciplines. They reported inconclusive results on learning and transfer: students performed well on assessments of knowledge of STELLA, but seemed less able to transfer their knowledge to other domains. A later study (Mandinach, 1989) reported that students, having experienced the systems thinking curriculum and used STELLA, were able to apply knowledge of systems concepts to scientific problems of varying complexity and sophistication, but concluded that additional curriculum development was needed.

• Miller et al. (1993) discussed a semi-quantitative modeling program called IQON. Semi-quantitative modeling differs from quantitative modeling in that variables might be ordered in terms of "positive," "low," or "increasing,"

and relationships might be characterized as "strong positive" or "weak immediate," rather than strictly quantitatively. (They also reported on several other tools they have developed which support quantitative and qualitative modeling.) Of interest in this study was whether these modeling tools could facilitate students' reasoning skills, both by using others' models and by creating their own. In a formative data analysis, they reported that students were able to use sophisticated causal reasoning in their models and were able to begin thinking about complex feedback systems.

• Schecker (1993) reported a study in which students enrolled in an eleventh grade mechanics course used STELLA in over one-fourth of the course lessons. Models were developed in class from scratch. Schecker posited that system dynamics modeling focuses on developing and testing a model (as opposed to simulation which focuses on exploring the consequences of varying initial values), and that modelling can help to accentuate the conceptual structure of a physical domain. He reported that it took at least 2 units before students were familiar with the software and systems thinking, after which they were able to work out model structures themselves in classroom discussions or work groups. He suggested (without empirical results) that icon-oriented modeling environments can help to accentuate the conceptual structure of a physical domain for students.

Studies on student-created models indicates that modeling can be made accessible to precollege students using carefully designed modeling environments such as Model-It, IQON, and STELLA. Most findings are preliminary; there are still many things to be learned about the place of modeling in the precollege curriculum.

Investigations of Writing Simulation Programs

Most of the literature exploring students writing simulation programs involves the question of what students learn by creating a model using a computer programming language. The Journal of Mathematical Behavior devoted an entire issue (1991, vol. 10, issue 1) to Boxer, a programming language which allows students to easily create simulation programs to solve problems involving motion. Feurzeig (1992) and Guzdial (1995) each described programming environments which allow students to write simulations by providing them with support for programming or for inquiry.

• diSessa (1991) reported on Boxer, a multipurpose computational medium allowing creation of hypertext, dynamic and interactive graphics, databases, and programs within a consistent, easily learned framework. Ploger (1991), diSessa, Abelson and Ploger (1991), and Adams and diSessa (1991) described various aspects of learning to program in Boxer. The Adams and diSessa article described students' investigations of a motion-simulating microworld. Anecdotal evidence showed that during the course of these investigations, students engaged in an interesting form of student-initiated learning--they learned to "cheat" at the simulation in order to solve difficult problems by directly programming objects in the microworld.

• Feurzeig (1992) described a visual programming language called Function Machines, which uses iconic representations of "programs." It was designed to support mathematical exploration and inquiry. The purpose of the study was to investigate the use and benefit of visualization tools on students' inquiry activities. He reported that appropriate computer modeling activities made the experience of doing science with Function Machines concrete and highly motivating for high school students. No empirical results on student learning are reported, however.

• A study by Guzdial (1995) described the creation of a scaffolded programming environment called EMILE that enables students to create physics models and run simulations with Hypercard without necessarily learning the Hypercard programming language. The software-realized scaffolding was designed support model-creation and student learning by eliciting articulation from, coaching, and communicating process to the learner. Guzdial reported that by creating models, students learned programming techniques and gained knowledge of physics concepts (e.g. velocity, acceleration, and projectile motion).

In summary, the few studies of student-programmed simulations reviewed indicate that students who use appropriately designed programming environments appear motivated, learn programming skills, gain conceptual knowledge, and engage in student-initiated learning.

Future directions

There is no firm consensus yet as to all the benefits which might accrue to learners as a result of running simulations or constructing models. Neither is there agreement as to which instructional strategies might make the most effective use of computer-based models. Many of the studies reviewed consisted of user testing reports,

case-based analyses, or formative research into the learning benefits of simulation and modeling. Most were conducted with high school students. Some involved a rather short treatment duration or were conducted outside of the science classroom in a clinical setting. There are examples, however, of promising, ongoing classroom-based research inquiry in which one study clearly built upon the work of another, notably the investigations using STELLA, GCK, and Boxer. The relative scarcity of research into computer-based models is probably due to several factors: the relatively recent introduction of the microcomputer into classroom use, the lack of generally available and easy-to-use models and modeling environments, the difficulty of collecting and analyzing data generated by students engaged in these kinds of activities, and the huge investment of time and effort involved in designing and implementing model-based software and research. Consequently, there are many possible fruitful directions for future research.

• First, a great deal remains to be understood about the ways in which students construct understandings of natural phenomena when engaged in making models or running simulations. For instance, in simulations, what kinds of simulation exploration strategies do students follow? What misconceptions might they form as a result of a simulation's necessary simplifications of reality? Do understandings of the simulation's representation of reality transfer to the real world, and if so, what circumstances promote such transfer? In modeling, what kinds of models can students really construct, and what can we infer from their models about their understanding of complex systems? What do students learn about a phenomena by creating a model of it, and what prior knowledge is necessary for the creation? In both modeling and simulation: What are the relationships between simulations, students' mental models, and student-constructed models? When interacting with computer-based models, what do students come to understand about the nature of models and the modeling process? What confidence do students place in the reliability and validity of simulations or of the models they themselves construct? Answers to these questions should inform software design, curriculum decisions, and pedagogical practices in science classrooms.

• Second, we need studies which investigate how simulations and modeling environments can be designed in order to effectively support student learning. Such studies might focus efforts upon interface and data representation techniques that prove to be the most useful, accessible and engaging to students and teachers. They would also help us understand how learning environments might accommodate diverse learning styles and preferences.

• Third, we need more large-scale, long-term studies of modeling and simulation in real classrooms in order to assess the influences of the various model-investigation activities on teacher attitudes and practices. Such implementations would help identify the problems that innovators of technology-based reforms encounter and solve.

• Fourth, we need investigations into the ways model-based inquiry affects students' attitudes toward science (Stratford & Finkel, 1995), their motivation for studying science, and their understanding of science as inquiry.

• Finally, we need studies which examine the relationship between data collected in real world and data generated by simulations or student-created models, in order to sort out issues of model accuracy, validation, and usefulness. For example, microcomputer-based laboratories (MBLs) might be coupled with computer simulations in an inquiry framework. This would help us with the larger question of how running simulations and building models might help students understand the role models play in authentic scientific inquiry and professional practice.

The field is wide open for exploring modeling and simulation in precollege classrooms, and preliminary results are promising. First, learner interactions with models of real world phenomena seem to provide opportunities for students to confront misconceptions and for researchers to learn more about students' problem solving skills and strategies. Second, running simulations and creating models appear to help students construct understandings of both science content knowledge and modeling process knowledge. Finally, systems thinking in general and system dynamics-based model-creation environments in particular may help students acquire useful conceptual knowledge and analysis skills.

References

Adams, S. T., & diSessa, A. (1991). Learning by "cheating": students' inventive ways of using a Boxer motion microworld. Journal of Mathematical Behavior, 10(1), 79-89.

American Association for the Advancement of Science (1993). Benchmarks for science literacy: Project 2061. New York: Oxford University Press.

Black, M. (1962). Models and metaphors: studies in language and philosophy. New York: Cornell University Press.

Brna, P. (1987). Confronting dynamics misconceptions. Instructional Science, 16, 351-379.

Brna, P. (1991). Promoting creative confrontations. Journal of Computers and Learning, 7, 114-122.

diSessa, A., Abelson, H., & Ploger, D. (1991). An overview of Boxer. Journal of Mathematical Behavior, 10 (1), 3-15.

Feurzeig, W. (1992). Visualization tools for model-based inquiry. Paper presented at the Conference on Technology Assessment, Los Angeles.

Feurzeig, W. Visualization in educational computer modelling.

Finkel, Elizabeth A., (1993). The Construction of Science in a High School Genetics Class. Unpublished doctoral dissertation, The University of Wisconsin-Madison.

Finkel, Elizabeth A., (1994, April). Making sense of genetics: students' knowledge use during problem solving in a high school genetics class. Paper presented at the Annual Meeting of the American Educational Research Association, New Orleans, LA.

Finkel, E.A., and Stewart, J., (1994). Strategies for model-revision in a high school genetics classroom. Mind, Culture, and Activity, 1(3), 168-195.

Forrester, J. W. (1968). Principles of Systems. Cambridge, MA: Wright-Allen Press.

Gee, B. (1978). Models as a pedagogical tool: can we learn from Maxwell? Physics Education, 13, 287-291.

Gilbert, S. W. (1991). Model building and a definition of science. Journal of research in science teaching, 28(1), 73-79.

Gorsky, P., & Finegold, M. (1992). Using computer simulations to restructure students' conceptions of force. Journal of Computers in Mathematics and Science Teaching, 11, 163-178.

Guzdial, M. (1995). Software-realized scaffolding to facilitate programming for science learning. Interactive Learning Environments, to appear.

Hafner, R. S. (1991). High school students' model-revising problem solving in genetics. Unpublished doctoral dissertation, University of Wisconsin, Madison.

High Performance Systems. (1992). STELLA II: an introduction to systems thinking: self-published.

Jackson, S., Stratford, S. J., Krajcik, J., & Soloway, E. (1995a). Making system dynamics modeling accessible to pre-college science students. Paper presented at the annual meeting of the American Educational Research Association, San Francisco, CA.

Jackson, S., Stratford, S. J., Guzdial, M., Krajcik, J., Soloway, E. (1995b). The ScienceWare Modeler: a case study of learner-centered design software. Paper presented at Working Conference on Technology Applications in the Science Classroom, The National Center for Science Teaching and Learning, Columbus, OH.

Jungck, J. R., Soderberg, P., Calley, J., Peterson, N., & Stewart, J. (1993). The BioQUEST Library. College Park, MD: University of Maryland Press.

Jungck, J. R., & Calley, J. (1985). Strategic simulations and post-Socratic pedagogy: constructing computer software to develop long-term inference through experimental inquiry. American Biology Teacher, 47, 11-15.

Kornblugh, M., & Little, D. (1976). The nature of a computer simulation model. Technological forecasting and social change, 9, 3-26.

Kruper, J., & Nelson, E. (1991). Developing and evaluating novel strategic computer simulations: a case study in translating an instructional pedagogy to the classroom. Paper presented at The International Conference on the Learning Sciences, Northwestern University, Evanston, IL.

Mandinach, E. B. (1988). The cognitive effects of simulations-modeling software and systems thinking on learning and achievement. Paper presented at the American Educational Research Association, New Orleans.

Mandinach, E. B. (1989). Model-building and the use of computer simulation of dynamic systems. Journal of Educational Computing Research, 5(2), 221-243.

Mandinach, E., & Cline, H. (1992, March). The impact of technological curriculum innovation on teaching and learning activities. Paper presented at the annual meeting of the American Educational Research Association, Boston, MA.

Mandinach, E. B., & Cline, H. F. (1994). Classroom dynamics: implementing a technology-based learning environment. Hillsdale, New Jersey: Lawrence Erlbaum Associates.

Miller, R., Ogborn, J., Briggs, J., Brough, D., Bliss, J., Boohan, R., Brosnan, T., Mellar, H., & Sakonidis, B. (1993). Educational tools for computational modelling. Computers in Education, 21(3), 205-261.

Ploger, D. (1991). Learning about the genetic code via programming: Representing the process of translation. The Journal of Mathematical Behavior, 10(1), 55-77.

Richards, J., Barowy, W., & Levin, D. (1992). Computer simulations in the science classroom. Journal of Science Education and Technology, 1(1), 67-79.

Roberts, N., Andersen, D. F., Deal, R. M., Garet, M. S., & Shaffer, W. A. (1983). Introduction to computer simulation: a system dynamics modeling approach. Portland, Oregon: Productivity Press.

Ronen, M., Langley, D., & Ganiel, U. (1992). Integrating computer simulations into high school physics teaching. Journal of Computer in Mathematics and Science Teaching, 11(3 & 4), 319-329.

Schecker, H. (1993). Learning physics by making models. Physics Education, 28, 102-106.

Silvert, W. (1993). Object-oriented ecosystem modelling. Ecological Modelling, 68, 91-118.

Simmons, P. E., & Lunetta, V. N. (1993). Problem-solving behaviors during a genetics computer simulation: beyond the expert/novice dichotomy. Journal of Research in Science Teaching, 30(2), 153-173.

Slack, S. J., & Stewart, J. (1990). High school students' problem-solving performance on realistic genetics problems. Journal of Research in Science Teaching, 27(1), 55-67.

Stewart, J., & Hafner, R. (1991). Extending the conception of "problem" in problem-solving research. Science Education, 75(1), 105-120.

Stewart, J., Hafner, R., Johnson, S., & Finkel, E. (1992). Science as model-building: computers and high-school genetics. Educational Psychologist, 27(3), 317-336.

Stratford, S. J., & Finkel, E. A. (1995). Impact of ScienceWare and Foundations on students' attitudes. Paper presented at the annual meeting of the National Association of Research in Science Teaching, San Francisco, CA.

White, B. (1993). ThinkerTools: causal models, conceptual change, and science education. Cognition and Instruction, 10(1), 1-100.

Acknowledgments

The author gratefully acknowledges the assistance of Joe Krajcik and Liza Finkel for the many helpful comments that they provided on drafts of this column. Supported by the National Science Foundation (RED 9353481) and the University of Michigan.